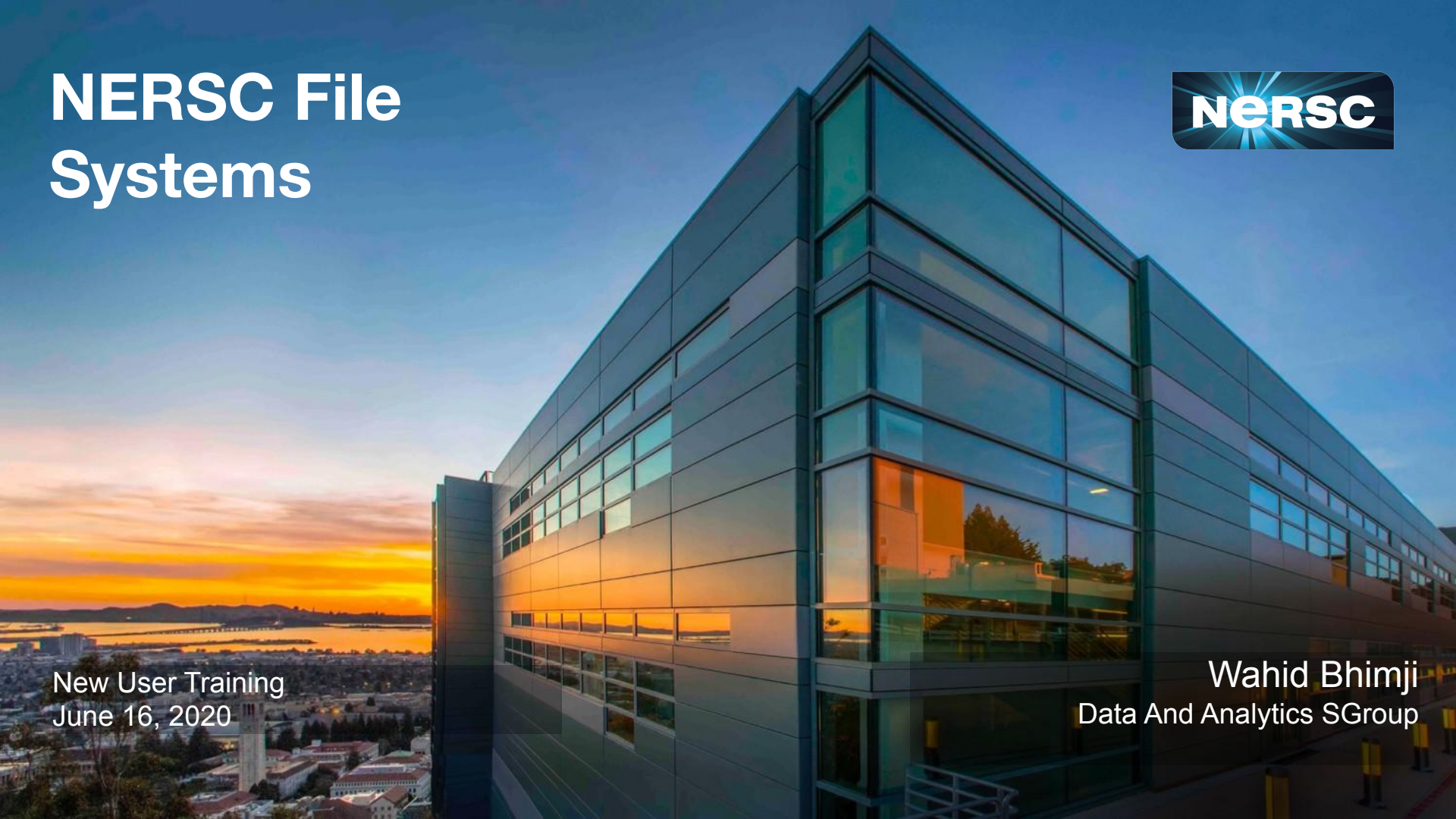
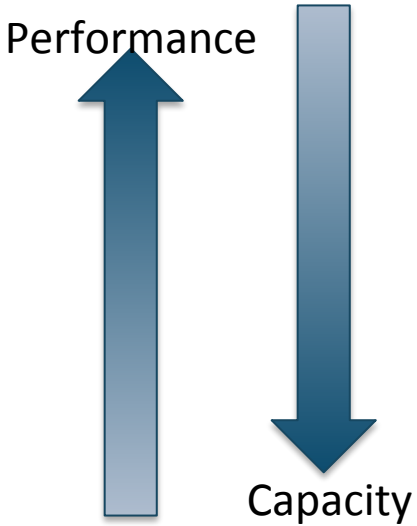
NERSC File Systems



New User Training
June 16, 2020

Wahid Bhimji
Data And Analytics SGroup

Simplified NERSC File Systems



Memory

Burst Buffer

Scratch

Community

HPSS

Global Common

Global Home

1.8 PB SSD Cori Burst Buffer
Cray Datawarp 1.8 TB/s,
temporary for job or campaign

28 PB HDD Cori Scratch
Lustre 700 GB/s,
temporary (12 wk purge)

57 PB HDD Community
Spectrum Scale (GPFS)
150 GB/s, permanent

150 PB Tape HPSS Archive
Forever

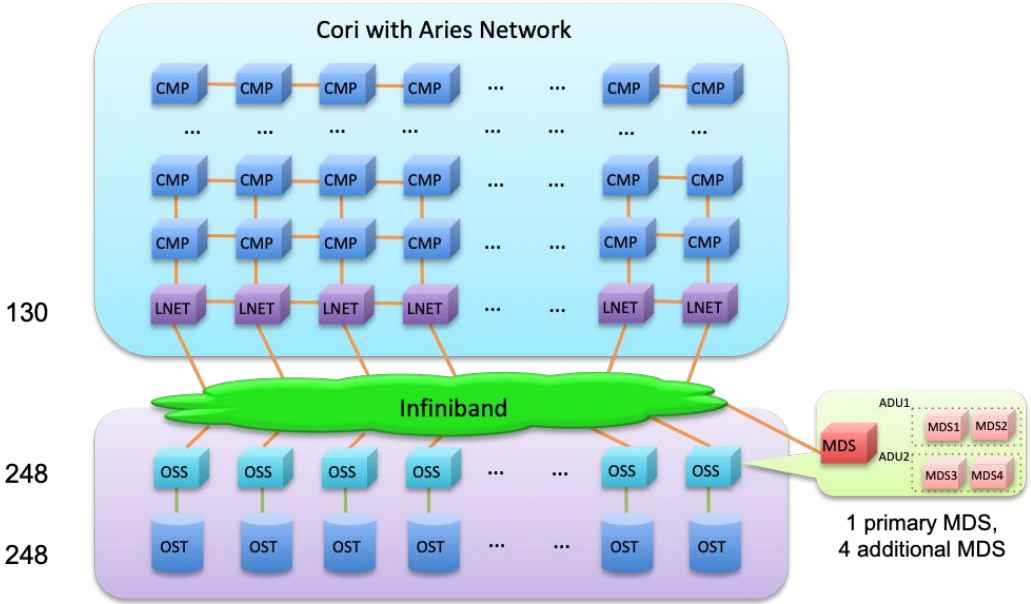
20 TB SSD Global Common Software
Spectrum Scale
Permanent
Faster compiling / Source Code

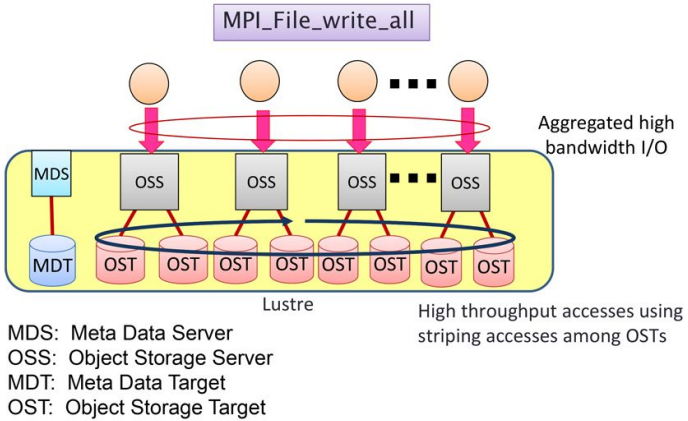
NERSC

Cori Scratch

Lustre, one of the most successful/mature HPC FS

Purged! Files not accessed for more than 12 weeks are automatically deleted





Using MPI-IO on Lustre[2]

1. https://en.wikipedia.org/wiki/Lustre_(file_system)
2. MPI-IO on Lustre: https://www.sys.r-ccs.riken.jp/ResearchTopics/fio/mpiio/

“Scratch”: Optimize Performance with Striping

Scratch Striping Recommendations

- By default data on 1 OST, ideal for small files and file-per-process IO
- Single shared file IO should be striped according to its size
- Helper scripts

  stripe_small, stripe_medium stripe_large
- Manually query with

  lfs getstripe <file_name>
- Manually set with

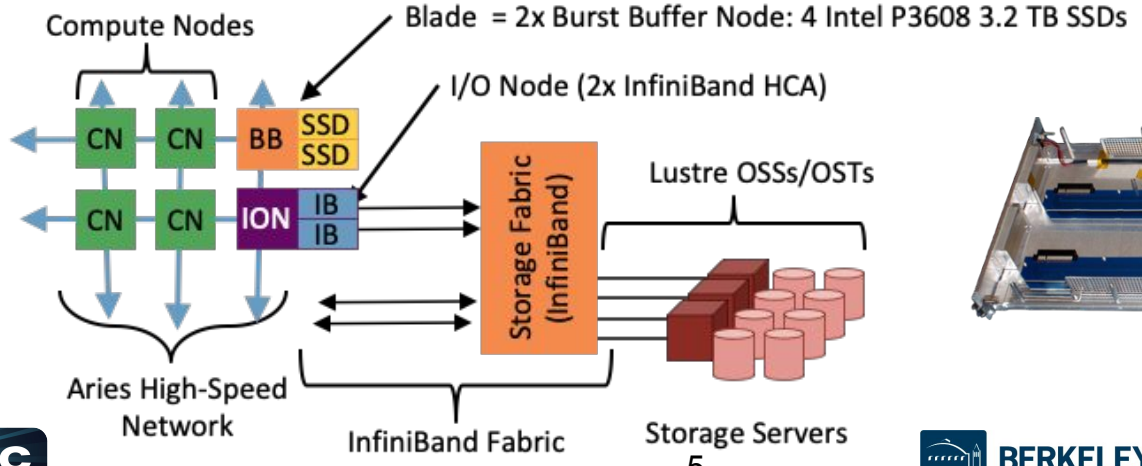
  lfs setstripe -S 1m -c 2 <empty_folder>

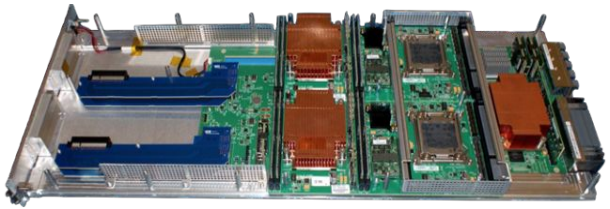
	Single Shared-File I/O	File per Process
File size (GB)	command	
< 1	keep default striping	keep default striping
1 - 10	`stripe_small`	keep default striping
10 - 100	`stripe_medium`	keep default striping
> 100	`stripe_large`	keep default striping
> 1000	`stripe_large`	`stripe_large`

Burst Buffer (BB)

Datawarp (DW): Cray's applications I/O accelerator

- For: Data read in/out by high IO-bandwidth or IOPS application
- Transient: Allocated per-job or per-campaign ('persistent') via SLURM integration
- Users see mounted POSIX filesystem. Striped across BB nodes.





Burst Buffer Example

```
#!/bin/bash
#SBATCH -q regular -N 10 -C haswell -t 00:10:00
#DW jobdw capacity=1000GB access_mode=striped type=scratch
#DW stage_in source=/global/cscratch1/sd/username/file.dat destination=$DW_JOB_STRIPED/ type=file
#DW stage_out source=$DW_JOB_STRIPED/outputs destination=/lustre/outputs  type=directory
srun my.x --infile=$DW_JOB_STRIPED/file.dat --outdir=$DW_JOB_STRIPED/outputs
```

- 'jobdw' – duration just for compute job (i.e. not 'persistent')
- 'access_mode=striped' – visible to all compute nodes, striped across BB nodes
  - Number of BB nodes depends on size requested - 'granularity' is 20 GB
- Data 'stage_in' before job start and 'stage_out' after
- `$DW_JOB_STRIPED` env variable points to the mountpoint
- Can also use interactively:

```
wbhimji@cori12:~> cat bbf.conf
#DW jobdw capacity=1000GB access_mode=striped type=scratch
wbhimji@cori12:~> salloc -q interactive -N 1 -C knl --time=00:30:00 --bbf=bbf.conf
```

Creating a Persistent Reservation on Burst Buffer

Can be used by any job (set unix file permissions to share)

Don't forget to delete the PR when not needed (and no more than 6 weeks after)

DW not for long term storage and not resilient - stage_out anything you cannot afford to lose

Create a PR

```
#!/bin/bash
#SBATCH -q regular -N 10 -C haswell -t 00:10:00
#BB create_persistent name=myBBname
capacity=1000GB access=striped type=scratch
```

Delete PR

```
#!/bin/bash
#SBATCH -q regular -N 10 -C haswell
-t 00:10:00
#BB destroy_persistent name=myBBname
```

Can check reservation outside job:

```
wbhimji@cori12:~> scontrol show burst

Name=wahid_test_apr15_2
CreateTime=2020-02-14T16:10:36 Pool=(null)
Size=61872MiB State=allocated
UserID=wbhimji(68441)
```

Use PR in your jobs

```
#SBATCH -q regular -N 10 -C haswell -t 00:10:00
#DW peristentdw name=myBBname
#DW stage_in
source=/global/cscratch1/sd/[username]/inputs
destination=$DW_PERSISTENT_STRIPED_myBBname
type=directory
srun my.x
--indir=$DW_PERSISTENT_STRIPED_myBBname/inputs
```

Community File System

- For: Large datasets that you need for a longer period
- Set up for sharing with group read permissions by default
- Not for intensive I/O - use Scratch instead
- Can share data externally by dropping it into a www directory
  Example: /global/cfs/cdirs/das/www/[username]
  https://portal.nersc.gov/project/das/[username]/
- Data is never purged. Snapshots. Usage is managed by quotas
- Projects can split their space allocations between multiple directories and give **separate** working groups **separate** quotas
  - Environment variable $CFS points to /global/cfs/cdirs

https://docs.nersc.gov/filesystems/community/

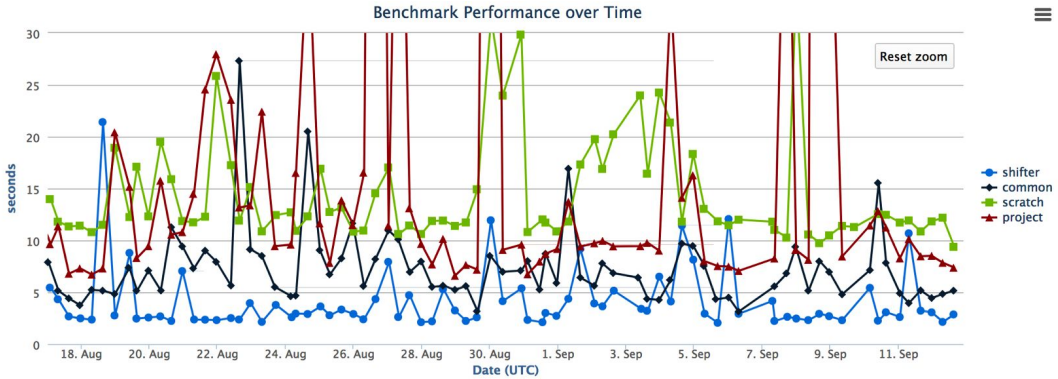
HPSS

- For: Data from your finished paper, raw data you might need in case of emergency, really hard to generate data
- HPSS is tape!
  - Data first hits a spinning disk cache and gets migrated to tapes
  - Files can end up spread all over, so use htar to aggregate into bundles of 100 GB - 2 TB
  - Archive the way you intend to retrieve the data

https://docs.nersc.gov/filesystems/archive/
https://docs.nersc.gov/filesystems/archive_access/

“Global Common”: Software Filesystem

- For: Software stacks - Why? Library load performance

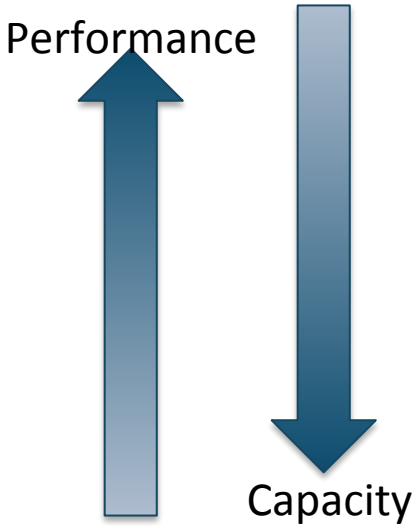


- Group writable directories similar to community, but with a smaller quota, /global/common/software/<projectname>
  - Write from login node; read-only on compute node
- Smaller block size for faster compiles than project

Home Directories

- For: Source files, compiling, configuration files
- 20G quota
- Not intended for intensive I/O (e.g. application I/O) - use Scratch instead
- Backed up monthly by HPSS
- Snapshots are also available e.g. my homedir is at /global/homes/.snapshots/2020-06-14/w/wbhimji

Simplified NERSC File Systems



Memory

Burst Buffer

Scratch

Community

HPSS

Global Common

Global Home

1.8 PB SSD Cori Burst Buffer
Cray Datawarp 1.8 TB/s,
temporary for job or campaign

28 PB HDD Cori Scratch
Lustre 700 GB/s,
temporary (12 wk purge)

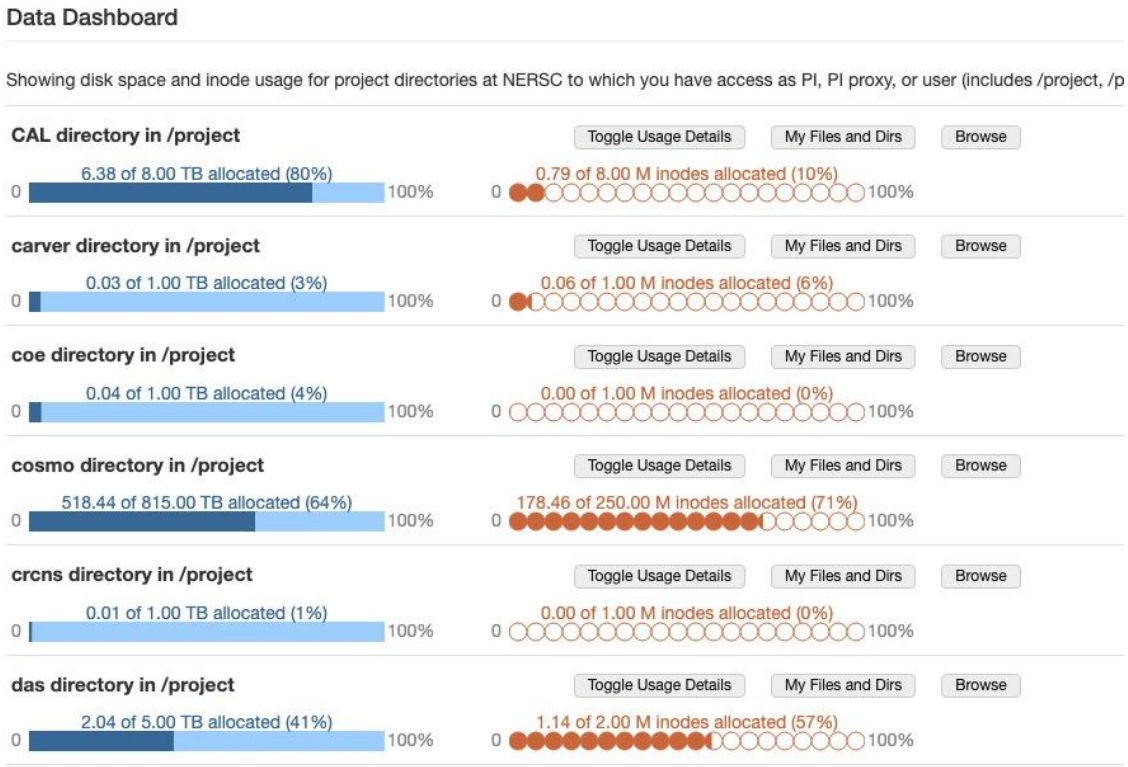
57 PB HDD Community
Spectrum Scale (GPFS)
150 GB/s, permanent

150 PB Tape HPSS Archive
Forever

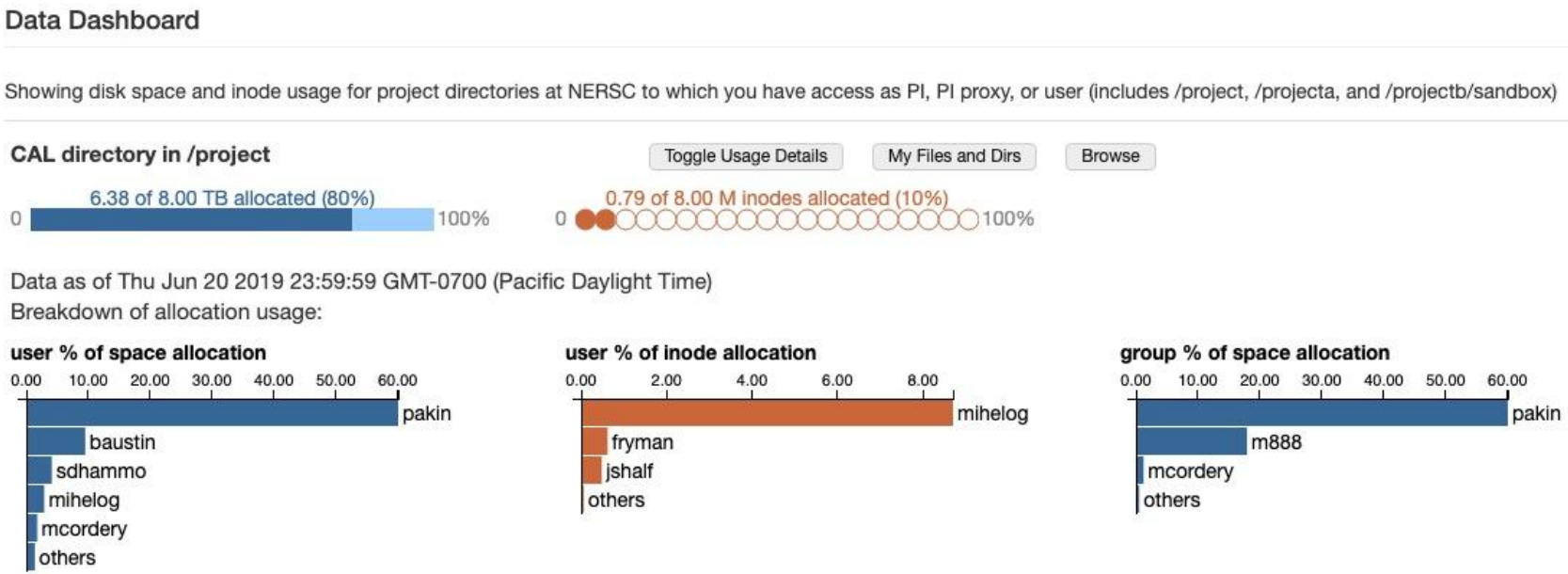
20 TB SSD Global Common Software
Spectrum Scale
Permanent
Faster compiling / Source Code

NERSC

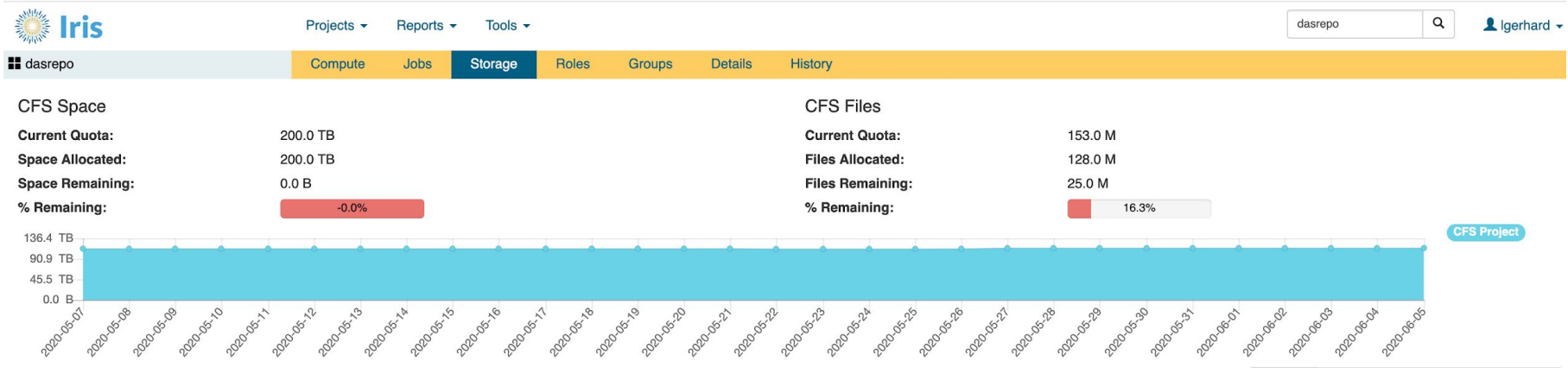
Data Dashboard



Data Dashboard: Usage Reports



Adjusting Quotas in IRIS



CFS Directory Usage

+ New | Edit | Rename | Activate

Directory	File System	Owner	Group	Active	Storage Used	Byte Limit	% Storage Used	Files Used	File Limit	% Files Used	Updated On
dasrepo	gpfs	Prabhat, Mr	dasrepo	✔	60.8 TB	90.0 TB	67.5%	27 M	100.0 M	26.6%	2020-06-05
ProjectDisCo	gpfs	Gerhardt, Lisa	projectd	✔	44.4 TB	91.0 TB	48.8%	134 K	20.0 M	0.7%	2020-06-05
mantissa	gpfs	Prabhat, Mr	mantissa	✔	5.2 TB	10.0 TB	51.8%	641 K	1.0 M	64.1%	2020-06-05
das	gpfs	Prabhat, Mr	das	✔	4.0 TB	5.0 TB	80.2%	1 M	2.0 M	59.7%	2020-06-05
ClimateNet	gpfs	Gerhardt, Lisa	climaten	✔	888.3 GB	1.0 TB	86.7%	108 K	1.0 M	10.8%	2020-06-05
datamap	gpfs	Gerhardt, Lisa	datamap	✔	111.4 GB	1.0 TB	10.9%	910 K	2.0 M	45.5%	2020-06-05
gbclimat	gpfs	Pseudo User, g…	gbclimat	✔	0.0 B	1.0 TB	0.0%	1	1.0 M	0.0%	2020-06-05
dastest	gpfs	Pseudo User, d…	dastest	✔	0.0 B	1.0 TB	0.0%	1	1.0 M	0.0%	2020-06-05

Resources

- Cori File Systems

  https://docs.nersc.gov/filesystems//
- NERSC Burst Buffer Web Pages

  https://docs.nersc.gov/filesystems/cori-burst-buffer/
- Example batch scripts

  https://docs.nersc.gov/jobs/examples/#burst-buffer
- DataWarp Users Guide

  https://pubs.cray.com/bundle/XC_Series_DataWarp_User_Guide_S-2558_publish_S2558_final/page/About_XC_Series_DataWarp_User_Guide.html


Thank You and
Welcome to
NERSC!
NERSC